Instalacja Rocky Linux z własnym nazewnictwem VG i LV

Typ: procedura **Dotyczy: Rocky Linux 9 / RHEL 9, instalator Anaconda**

Czas: ~20 min

Efekt: system zainstalowany z własnym nazewnictwem grup wolumenów i wolumenów logicznych — osobna grupa na każde przeznaczenie, zamiast domyślnej `rl-root` / `rl-swap` generowanej przez instalator.

Skróty użyte na tej stronie (6)

Skrót	Co znaczy
GRUB	Grand Unified Bootloader: program, który jako pierwszy startuje po włączeniu komputera i podaje jądru, co ma uruchomić
GUI	Graphical User Interface: okienka i myszka zamiast komend
LV	Logical Volume: wolumen logiczny w LVM, na nim leży system plików
LVM	Logical Volume Manager: warstwa w Linuksie składająca dyski w elastyczne wolumeny
RHEL	Red Hat Enterprise Linux: komercyjna dystrybucja Linuksa
VG	Volume Group: grupa wolumenów w LVM, pula złożona z dysków

Zanim zaczniemy

Z ŻYCIA WZIĘTE

O 8:30 Tomek instaluje nowy serwer bazy danych, który ma zastąpić starzejący się pg01. Zamiast klikać przez GUI instalatora, wchodzi w terminal Anacondy i ręcznie zakłada osobne grupy wolumenów dla systemu, `/var` i danych PostgreSQL, żeby uniknąć domyślnego `rl-root`, które trzy miesiące później i tak trzeba by przemianować. Dzięki temu powiększenie

> wolumenu z danymi pół roku później to jedna komenda `lvextend`, a nie ryzykowna operacja na wolumenie systemowym.

Dyski muszą być podpięte i widoczne przed uruchomieniem instalatora. Partycjonowanie wykonujemy z sesji terminalowej instalatora, a nie z GUI — dzięki temu mamy pełną kontrolę nad nazwami.

Wymaganie	Uwaga
Nośnik instalacyjny Rocky Linux 9	również przez iLO / iDRAC / konsolę wirtualną
Wszystkie dyski podpięte	dołożenie dysku po instalacji wymaga ponownego partycjonowania
Rozplanowane przeznaczenia	jedna grupa wolumenów na jedno przeznaczenie

Konwencja nazewnicza: `vg<Przeznaczenie>` oraz `lv<Przeznaczenie>`. Nazwa wolumenu logicznego powtarza przeznaczenie grupy, dzięki czemu pełna ścieżka czyta się sama: `/dev/mapper/vgPostgres-lvPostgres`.

Przeznaczenie	Grupa wolumenów	Wolumen logiczny
system operacyjny	`vgOS`	`lvOS`
`/var`	`vgVAR`	`lvVAR`
baza PostgreSQL	`vgPostgres`	`lvPostgres`
baza Oracle	`vgDB`	`lvOracle`

Uważaj: układ zapisany na dysk w trakcie instalacji jest praktycznie nieodwracalny bez reinstalacji. Zmiana nazw po instalacji jest możliwa, ale wymaga przebudowy initramfs i poprawek w GRUB — pomyłka kończy się systemem, który nie startuje.

Przebieg

1. Uruchomienie instalatora i przejście do terminala

Po uruchomieniu instalatora wybieramy odpowiedni język i wciskamy kombinację klawiszy:

```
CTRL + ALT + F3
```

Przeniesie nas to do sesji terminalowej, w której będziemy mogli dowolnie zarządzać dyskami.

2. Weryfikacja wykrytych dysków

Sprawdzamy, jakie dyski widzi system:

```
lsblk
```

Przykład:

```
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
sda      8:0    0   100G  0 disk
sdb      8:16   0   200G  0 disk
sdc      8:32   0   500G  0 disk
sr0     11:0    1   9,5G  0 rom
```

3. Utworzenie wolumenów fizycznych

Na początku musimy utworzyć physical volume poprzez komendę `pvcreate`:

```
pvcreate /dev/sdX
```

> (gdzie `/dev/sdX` to nazwa urządzenia blokowego z wyniku `lsblk`)

Tworzymy resztę wymaganych urządzeń fizycznych:

```
pvcreate /dev/sda
pvcreate /dev/sdb
pvcreate /dev/sdc
```

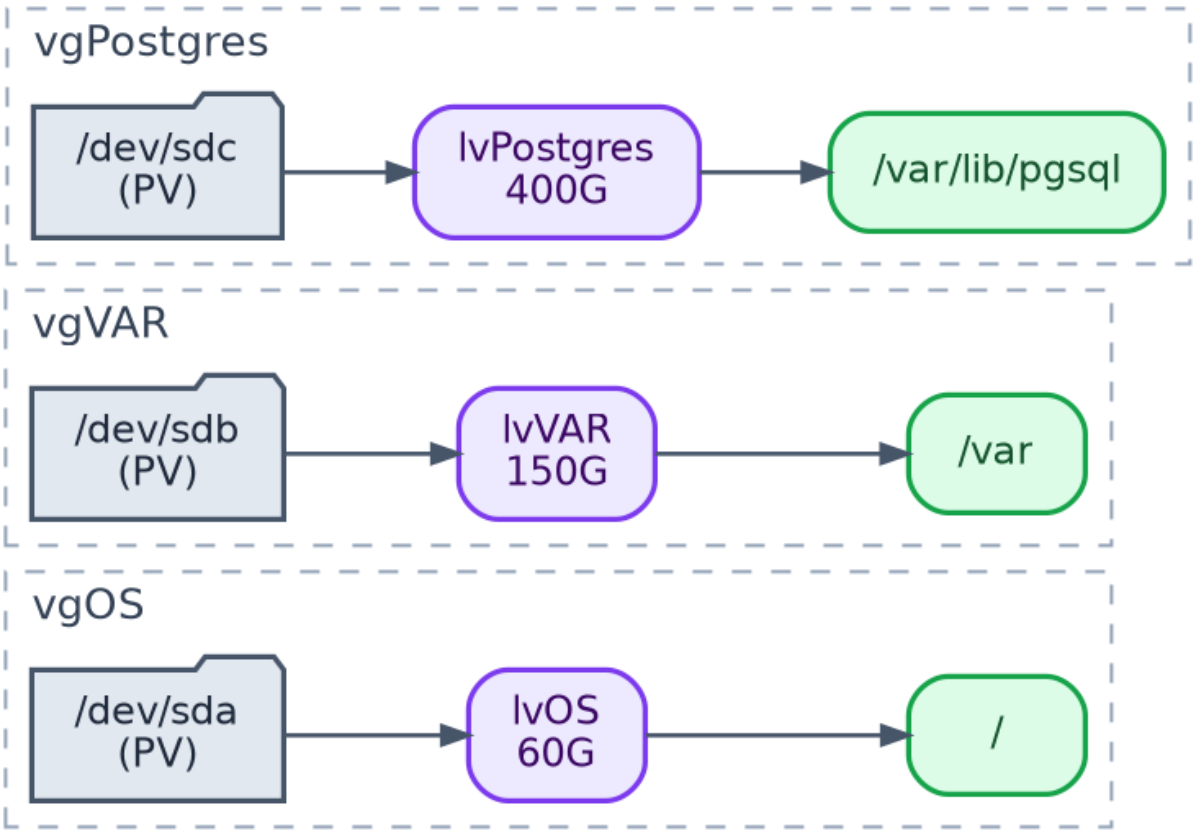
Weryfikujemy:

```
pvs
```

Przykład:

```
  PV         VG Fmt  Attr PSize    PFree
  /dev/sda      lvm2 ---  <100.00g <100.00g
  /dev/sdb      lvm2 ---  <200.00g <200.00g
  /dev/sdc      lvm2 ---  <500.00g <500.00g
```

4. Utworzenie grup wolumenów



Każdy dysk fizyczny staje się osobnym wolumenem fizycznym, trafia do własnej grupy wolumenów, a z niej powstaje jeden wolumen logiczny montowany pod właściwą ścieżką (kliknięcie otwiera pełny rozmiar)

Tworzymy osobną grupę na każde przeznaczenie:

```
vgcreate vgOS /dev/sda
```

```
vgcreate vgVAR /dev/sdb
```

```
vgcreate vgPostgres /dev/sdc
```

Weryfikujemy:

```
vgs
```

Przykład:

```
  VG         #PV #LV #SN Attr   VSize    VFree
  vgOS         1   0   0 wz--n- <100.00g <100.00g
  vgPostgres   1   0   0 wz--n- <500.00g <500.00g
  vgVAR        1   0   0 wz--n- <200.00g <200.00g
```

5. Utworzenie wolumenów logicznych

Wolumen systemowy — zostawiamy zapas w grupie na późniejsze powiększenie:

```
lvcreate -L 60G -n lvOS vgOS
```

Wolumen na `/var`:

```
lvcreate -L 150G -n lvVAR vgVAR
```

Wolumen na dane bazy:

```
lvcreate -L 400G -n lvPostgres vgPostgres
```

> Nie wykorzystujemy `100%FREE` przy zakładaniu. Wolne miejsce w grupie jest tym, co pozwala później powiększyć wolumen jedną komendą, bez dokładania dysku.

Weryfikujemy:

```
lvs
```

Przykład:

```
  LV          VG         Attr       LSize   Pool Origin Data%  Meta%
  lvOS        vgOS       -wi-a-----  60.00g
  lvPostgres  vgPostgres -wi-a----- 400.00g
  lvVAR       vgVAR      -wi-a----- 150.00g
```

6. Powrót do instalatora i przypisanie punktów montowania

Wracamy do instalatora kombinacją:

```
CTRL + ALT + F6
```

Wchodzimy w *Installation Destination* → *Custom* → *Done*. Utworzone wolumeny są już widoczne na liście po lewej stronie. Każdemu przypisujemy punkt montowania:

Wolumen	Punkt montowania	System plików
`vgOS-lvOS`	`/`	xfs
`vgVAR-lvVAR`	`/var`	xfs
`vgPostgres-lvPostgres`	`/var/lib/pgsql`	xfs

Partycję `/boot` i `/boot/efi` zostawiamy poza LVM — instalator utworzy je na dysku systemowym.

7. Instalacja

Zatwierdzamy układ i uruchamiamy instalację. Nazwy grup i wolumenów zostają takie, jakie utworzyliśmy.

Weryfikacja

Po pierwszym uruchomieniu systemu sprawdzamy, czy nazewnictwo się utrzymało:

```
lsblk
```

Przykład:

```
NAME                    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
sda                       8:0    0   100G  0 disk
├─sda1                    8:1    0     1G  0 part /boot
└─sda2                    8:2    0    99G  0 part
  └─vgOS-lvOS           253:0    0    60G  0 lvm  /
sdb                       8:16   0   200G  0 disk
└─vgVAR-lvVAR           253:1    0   150G  0 lvm  /var
sdc                       8:32   0   500G  0 disk
└─vgPostgres-lvPostgres 253:2    0   400G  0 lvm  /var/lib/pgsql
```

Sprawdzamy wolne miejsce w grupach — to jest zapas na powiększanie:

```
vgs
```

Przykład:

```
  VG         #PV #LV #SN Attr   VSize    VFree
  vgOS         1   1   0 wz--n-  <99.00g  39.00g
  vgPostgres   1   1   0 wz--n- <500.00g 100.00g
  vgVAR        1   1   0 wz--n- <200.00g  50.00g
```

Sprawdzamy montowania:

```
df -h
```

Wycofanie zmian

Jeśli system jest już zainstalowany z domyślnym nazewnictwem, nazwy da się zmienić na żywo — ale procedura dla wolumenu z systemem jest ryzykowna i opisana osobno na stronie o zmianie nazw VG i LV.

Zmiana nazwy grupy, która **nie** zawiera roota ani swapa, jest bezpieczna:

```
vgrename StaraNazwa vgPostgres
```

```
lvrename vgPostgres StaraNazwaLV lvPostgres
```

Poprawiamy wpis w `/etc/fstab` i montujemy:

```
mount -a
```

Gdy coś nie zadziała

Objaw	Przyczyna	Naprawa
Instalator nie widzi utworzonych grup	partycjonowanie wykonane po wejściu w *Installation Destination*	wychodzimy z okna i wchodzimy ponownie — lista odświeża się przy wejściu
`Device /dev/sdX not found` przy `pvcreate`	dysk niepodpięty albo widoczny pod inną nazwą	`lsblk` i `lsscsi`, sprawdzamy rzeczywistą nazwę

Objaw	Przyczyna	Naprawa
`Device /dev/sdX excluded by a filter`	na dysku jest stara sygnatura LVM lub tablica partycji	`wipefs -a /dev/sdX` i ponownie `pvcreate`
Po instalacji grupa nazywa się `rl`	układ został nadpisany przez automatyczne partycjonowanie	reinstalacja z *Custom*, albo zmiana nazw po instalacji
System nie startuje po zmianie nazw	initramfs i GRUB wskazują na stare nazwy	tryb ratunkowy, `dracut -f` i `grub2-mkconfig`

Powiązane: LVM: dysk fizyczny, grupa wolumenów, wolumen logiczny · Powiększanie przestrzeni dyskowej bez restartu · Zmiana nazwy grupy wolumenów i wolumenu logicznego · Montowanie: mount, /etc/fstab, zasoby sieciowe

Źródła: `man 8 pvcreate` · `man 8 vgcreate` · `man 8 lvcreate` · dokumentacja instalatora Anaconda

---

Revision #19
Created 2025-04-01 09:55:59 UTC by Tomasz Konieczny
Updated 2026-10-03 19:09:27 UTC by Tomasz Konieczny